Modelo de Calidad

# Algoritmo para obtención del nivel de calidad alcanzado

### Características y Subcaracterísticas Utilizadas:

1. Funcionabilidad
   1. Seguridad de Acceso
   2. Exactitud de los resultados
2. Fiabilidad
   1. Tolerancia a fallos
   2. Capacidad de recuperación de errores
3. Mantenibilidad
   1. Capacidad del código de ser analizado
   2. Capacidad del código de ser cambiado
   3. Estabilidad
4. Usabilidad
   1. Capacidad de ser entendido
   2. Capacidad de ser operado
   3. Capacidad de ser atractivo para el usuario
5. Eficiencia
   1. Utilización de recursos
   2. Comportamiento frente al tiempo
6. Portabilidad
   1. Adaptabilidad
   2. Instalabilidad

## Niveles de Puntuación

Se puntuara cada subcaracterística como mala, regular o buena, dependiendo de los criterios que se mencionarán a continuación. Luego se realizara un promedio con los puntajes anteriores para obtener un valor general de la característica. El listado anterior se encuentra ordenado de menor a mayor peso, que se utilizara para ponderar cada promedio obtenido.

El promedio general de toda la evaluación determinara el nivel de calidad obtenido dentro de las siguientes clasificaciones:

* SATISFACTORIO
* MINIMAMENTE ACEPTABLE
* NO SATISFACTORIO

## Criterios de evaluación

1. Funcionabilidad:
   1. Seguridad de Acceso: Capacidad del producto software para asegurar la integridad de los datos y la confidencialidad de estos.

* Características a medir:
  + Encriptación de datos
  + Inicio de sesión de usuarios
* Evaluación:
  + Mala [0] No cumple con alguna característica.
  + Regular [1] Cumple con 1 característica.
  + Buena [2] Cumple con 2 características.
* Puntuación:
  1. Exactitud de los resultados: Es la capacidad del producto software para proporcionar los resultados con el grado necesario de precisión.
* Características a medir:
  + Grado de precisión en el resultado dado.
* Evaluación:
  + Mala [0]: Los resultados tienen un error del orden de 10-3 o superior.
  + Regular [1-2]: Los resultados tienen un error del orden entre10-4y 10-6.
  + Buena [3]: Los resultados tienen un error del orden de10-7 o inferior
* Puntuación:

1. Fiabilidad:
   1. Tolerancia a fallos: Es la capacidad del producto software de mantener la integridad de los datos cuando se producen fallas del sistema.

* Características a medir:
  + Cuando sucede un error se protegen los datos procesados.
  + Se realiza un log de actividades que el sistema estaba haciendo.
  + Muestra por pantalla el mensaje de error
* Evaluación:
  + Mala [0]: No cumple con ninguna característica
  + Regular [1-2]: Cumple con una o dos características
  + Buena [3]: Cumple con las tres características
* Puntuación:
  1. Capacidad de recuperación de errores: Es la capacidad del sistema de reanudar sus actividades cuando se producen errores críticos.
* Características a medir:
  + El sistema reanuda las actividades si se produce una falla crítica.
  + Reanuda sus actividades y vuelve al estado en que estaba.
* Evaluación:
  + Mala [0]: No cumple con ninguna característica
  + Regular [1]: Cumple con una o dos características
  + Buena [2]: Cumple con las tres características
* Puntuación:

1. Mantenibilidad:
   1. Capacidad del código de ser analizado: Para evaluar la capacidad que tiene el código para ser analizado se tiene en cuenta el porcentaje de comentarios que posee el código por cada método y en general.

* Evaluación:
  + Mala [0]: 14% o menos del código comentado.
  + Regular [1-2]: Entre 15 y 29% del código comentado.
  + Buena [3]: 30% o más del código comentado.
* Puntuación:
  1. Capacidad del código de ser cambiado: Para evaluar la capacidad que tiene el código para ser cambiado se tomará en cuenta la complejidad ciclomática del método.
* Características a medir:
  + Complejidad ciclomática del código
* Evaluación:
  + Mala [0]: La complejidad ciclomática es mayor o igual a 21.
  + Regular [1]: La complejidad ciclomática está entre 10 y 21.
  + Buena [2]: La complejidad ciclomatica es menor a 10
* Puntuación:
  1. Estabilidad: Para determinar la estabilidad del software se evalúa el promedio de fallas que presenta el producto por prueba.
* Características a medir:
  + Se realizaron los siguientes lotes de prueba:

1. Nombre del paciente de formato valido = letras y espacio.
2. Nombre del paciente de formato invalido = número y caracteres distintos del espacio.
3. Dni de formato valido = números.
4. Dni de formato invalido = letras o caracteres.
5. Fecha de formato valido = números y carácter separador “-“
6. Fecha de formato invalido = letras y otros caracteres que no sean “-”.
7. Tipo de sangre de formato valido = letras mayúscula A, B y O y los caracteres “+” o “-”.
8. Tipo de sangre de formato invalido = letras minúsculas, números y caracteres distintos de “+” o “-”.
9. Teléfono de formato valido = números.
10. Teléfono de formato invalido = letras u otros caracteres.
11. Calle de la dirección de formato valido = letras y espacio.
12. Calle de la dirección de formato invalido = números y otros caracteres distintos del espacio.
13. Número de la dirección de formato valido = números.
14. Número de la dirección de formato invalido = letras y caracteres.
15. Código de los pacientes de formato valido = números.
16. Código de los pacientes de formato invalido = letras y caracteres.
17. Para los médicos se repiten las mismas pruebas.

* Evaluación:
  + Mala [0]: El software presenta 10 o más errores por prueba.
  + Regular [1-2]: El software presenta entre 5 y 9 errores por prueba.
  + Buena [3]: El software presenta menos de 5 por prueba.
* Puntuación:

1. Usabilidad:
   1. Capacidad de ser entendido: Capacidad que posee el software, para ayudar a los usuarios ante una determinada situación donde se necesite asistencia.

* Características a medir:
  + Posee ayuda contextual sobre menús y botones de acción.
  + Manual de usuario incorporado al sistema como un menú dedicado.
* Evaluación:
  + Mala[0]: No cumple con alguna característica
  + Regular [1]: Cumple con 1 característica.
  + Buena [2]: Cumple con 2 características.
* Puntuación:
  1. Capacidad de ser operado: Es la capacidad del producto software de ser utilizado sin asistencia adicional. Se valúa qué requiere el usuario para operar correctamente el producto.
* Evaluación:
  + Mala [0]: El usuario requiere consultar a personal especializado para operar el producto software.
  + Regular [1]: El usuario requiere ayuda contextual y manual de uso para operar el producto software.
  + Buena [2]: El usuario opera el producto software sin asistencia.
* Puntuación:
  1. Capacidad de ser atractivo para el usuario: Es la agrupación correcta de funcionalidad del producto software en su interfaz gráfica, desde su agrupación lógica hasta el número promedio de pasos para alcanzar una función o contenido específico.
* Evaluación:
  + Mala [0]: seis o más pasos promedio sin organización de categoría.
  + Regular [1]: Entre 3 y 5 pasos promedio y distribuidos en categorías.
  + Buena [2]: 1 o 2 pasos promedio y distribuidos en categorías.
* Puntuación:

1. Eficiencia:
   1. Utilización de recursos: Se evaluará la eficiencia del producto software de acuerdo al porcentaje de uso de procesador que realice.

* Evaluación:
  + Mala [0]: 41% o más de uso de procesador.
  + Regular [1]: 11% a 40% de uso de procesador.
  + Buena [2]: 10% o menos de uso de procesador.
* Puntuación:
  1. Comportamiento frente al tiempo: Se evaluará el tiempo que está el producto software sin informarle al usuario del estado en que se encuentra la solicitud que realizó.
* Evaluación:
  + Mala [0]: El producto está 5 o más segundos sin informar al usuario del estado de la solicitud.
  + Regular [1]: El producto está entre 2 y 4 segundos sin informar al usuario del estado de la solicitud.
  + Buena [2]: El producto está menos de 1 segundo sin informar al usuario del estado de la solicitud.
* Puntuación:

1. Portabilidad:
   1. Adaptabilidad: Es la capacidad del producto software de adaptarse a diferentes sistemas operativos sin cambiar su estructura interna.

* Características a medir:
  + Cantidad de sistemas operativos con los que es compatible
* Evaluación:
  + Mala[0]: Es compatible con solo un sistema operativo
  + Regular [1]: Es compatible con dos sistemas operativos
  + Buena [2]: Es compatible con tres o más sistemas operativos
* Puntuación:
  1. Instalabilidad: El producto software debe poder ser instalado en una cantidad mínima de pasos.
* Características a medir:
  + Cantidad de pasos para instalar el producto
* Evaluación:
  + Mala[0]:Se instala en más de diez y siete pasos
  + Regular [1]: Se instala entre seis y tres pasos
  + Buena [2]: Se instala en uno o dos pasos
* Puntuación:

## Resultados

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Característica | Promedio puntuación | | Ponderación | Puntaje general |
| Portabilidad |  | | 2 |  |
| Eficiencia |  | | 3 |  |
| Usabilidad |  | | 4 |  |
| Mantenibilidad |  | | 5 |  |
| Fiabilidad |  | | 6 |  |
| Funcionabilidad |  | | 7 |  |
| **TOTAL:** | |  |

|  |  |
| --- | --- |
| NIVELES DE ACEPTACION | PUNTAJE DE ACEPTACION |
| SATISFACTORIO | >50 |
| MINIMAMENTE ACEPTABLE | Entre 35 y 50 |
| NO SATISFACTORIO | <35 |